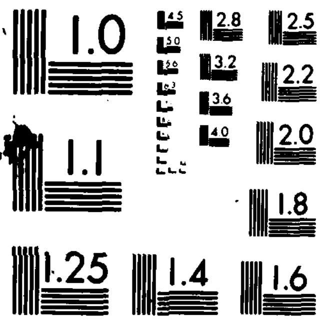
AD-A113 388 NAVAL RESEARCH LAB WASHINGTON DC F/G 9/2
AN INTELLIGENT CONTROL STRATEGY FOR COMPUTER CONSULTATION.(U)
APR 82 J SLAGLE, E HALPERN
UNCLASSIFIED NRL-MR-4789 NL

END
DATE
FILMED
4 82
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

AD A113388

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NRL Memorandum Report 4789	2. GOVT ACCESSION NO. AD-A113 388	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) AN INTELLIGENT CONTROL STRATEGY FOR COMPUTER CONSULTATION		5. TYPE OF REPORT & PERIOD COVERED Interim report on a continuing NRL problem. 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) J. Slagle and E. Halpern		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Research Laboratory Washington, DC 20375		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 75-0230-A-2
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE April 8, 1982 13. NUMBER OF PAGES 78
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Artificial intelligence
Rule-based system
Expert consultant system
Inference network
Best-first control strategy
Bayes rule

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The heart of a computerized expert system such as MYCIN or PROSPECTOR, is embodied in an inference network of associated propositions. Inference networks are organized with rules that direct the propagation of information. A top proposition under consideration is updated by depth-first traversal of the entire network in many current systems. We have applied the concept of "merit" to consultant systems to direct this

(Continues)

cont A

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. ABSTRACT (Continued)

network traversal. Merit values force a limited best-first traversal of the inference network, requesting only the most pertinent information from the user. The notion of merit previously used in the MULTIPLE system has been extended to the MYCIN style of propagation and to the subjective Bayesian updating used in PROSPECTOR. In addition, we propose a general mechanism for merit calculation with any differentiable updating scheme. Merits will provide an intelligent, efficient mechanism for controlling new, more versatile expert consultant systems.

CONTENTS

AN INTELLIGENT CONTROL STRATEGY FOR COMPUTER CONSULTATION

1. Introduction To Inference Networks

Expert consultant systems are presently available for various classes of problems. The most well known of these systems is MYCIN which has demonstrated a high level of competence in the diagnosis of infectious diseases. MYCIN operates on a system of inexact reasoning, propagating new information through an inference network in the form of certainty factors [5,6]. Various other expert systems, such as EMYCIN [3], have been cast in the mold of MYCIN.

The essence of an expert consultant is embodied in a graph called the inference network. Nodes on this network are representations of individual propositions, describing parameters relevant to the particular problem under study. Links connecting these propositions stipulate mathematical functions, combining antecedent propositions to update a consequent. These links or rules, as they are often called, define implications directed from antecedent to consequent, organizing the network to allow propagation of information. The inference network may have a simple tree structure, with each proposition acting as the antecedent of only one other proposition, or it may have a more complicated graph structure in which one antecedent has several consequents. We have designed and implemented our inferencing systems with acyclic networks to avoid indefinite looping during propagation. This restriction, though it greatly simplified the current design, should not be an absolute requirement, since a related work [11] did provide for cycles.

Manuscript submitted February 10, 1982.

Most inference networks will contain a limited number of nodes, typically consequents that imply no other proposition, that are the subjects of the inference evaluation process. Let us refer to these propositions as top propositions or consequents. In an inference network designed to predict the probability of rain, there may be one top proposition representing the chance of rain. A more complex problem involving several competing hypotheses may require several top propositions simultaneously. These top propositions may have independent inference networks, or they may share antecedent propositions. In general, they will have some common antecedents and other antecedents specific to each consequent, resulting in a complicated graph structure for the inference network.

Propositions on the inference network of a consulting system will be classified as "askable" or "unaskable". Askable propositions are those which the user may be reasonably expected to supply. Unaskable propositions are those more esoteric concepts whose resolution we prefer to leave to the system. Often it may be reasonable to associate a degree of askability with each askable proposition. A knowledgeable user may save time by responding to propositions of low askability. Requesting the same information from a less experienced user, however, may be a complete waste of time.

Top propositions are nearly always classified as unaskable. When the user provides the information requested for askable propositions elsewnere on the network, that information may be propagated toward the top propositions. The most common technique

applied by expert systems for updating the top propositions is a depth-first traversal of the inference network with reverse chaining of the rules. When an askable node is traversed, the user is prompted for the respective parameter. Once the user supplies the requested information, his response is propagated, and the traversal continues. If, however, the user is unable to update the parameter, the traversal is expanded to the antecedents of that unanswered proposition. This depth-first reverse chaining mechanism thus expands from a consequent to its antecedents in a direction opposite to that specified by the links, and then propagates back the information in the manner specified by the implications when it returns.

The most time-consuming aspect of expert computer consultation is the dialogue required between the user and the system to provide the propositional parameters. Since the consultation time is roughly proportional to the time spent responding to questions, an expert system may be considerably more efficient when its thirst for data is restricted. An intelligent system, asking the most pertinent questions first, and avoiding irrelevant propositions, will react much like a human consultant. Such a system would save an enormous amount of time by avoiding many of the propositions traversed in a classical depth-first approach.

The greatest obstacle to the development of such an intelligent expert system is the need for a general mechanism to choose the most appropriate questions in the network. We propose the

utilization of merits as developed for MULTIPLE [7,8,9,10] to provide this mechanism. Merits will guide the traversal of inference networks with a best-first strategy. The most pressing questions are asked first, and all questioning is terminated when there remains no chance of significantly altering the top proposition. Furthermore, this mechanism will apply to any type of inference link that can be expressed as a differentiable function.

2. Other Intelligent Expert Systems.

Certain simple techniques for pruning the depth-first traversal of inference networks have been proposed for various systems. These methods generally eliminate the traversal of nodes already proved to be true or false. For example, assume that a consequent H is true if either E1 or E2 is true. If we have already found E1 to be true, and have no other reasons for desiring to know the value of E2, then we may prune off E2; H has been proved regardless of the status of E2. Similarly, if H is true only when both E1 and E2 are true, and we know E1 to be false, there is no need to work on E2.

A more sophisticated design for ordering a depth-first traversal is presented in PROSPECTOR [2]. The MARK IV control strategy will first select a top proposition and then attempt to select the antecedent most likely to influence that proposition. In selecting an appropriate antecedent for questioning, a function, the J* function, is evaluated for each rule linking an antecedent to the current proposition. The antecedent with the greatest J* function value is selected for questioning. The J* function operates by combining four considerations: extreme strengths of the rule, the current strength of the rule, the prior probability of the antecedent, and the measure of belief or disbelief in the consequent. This mechanism results in a complicated, apparently ad hoc solution for ordering the depth first traversal.

The algorithm presented by PROSPECTOR for ordering the depth-first traversal provides a good start in working toward an intelligent expert system. Their system however, suffers from several basic constraints: They attempt to optimize their inference network traversal within the framework of a depth-first traversal. This constricts the pathways they must follow through the inference network. Once a node is traversed, the depth-first mechanism will never return to that part of the network. Furthermore, the optimization provided by the J* function with the MARK IV control strategy is local to the sons of a single node. Just because a proposition is the best son of the node being considered, there is no guarantee that it will also be an optimal proposition to work on when the entire inference network is considered. A more advanced control strategy might search for the globally optimal proposition in the entire inference network, and question the user on that item. Such a technique is proposed in this paper, and compared to the PROSPECTOR control strategy in sections 8-10.

The CASNET (causal-associational network) system attempts to approach this control strategy dilemma from the viewpoint of finding the next best node to work on in a global sense. CASNET assigns each proposition in its network a weight corresponding to the presence of evidence in support of that proposition [12]. The system considers both forward and reverse weights corresponding to the plausibility of a node as determined by its antecedents and consequents, respectively. A combined weight, actually the maximum of the forward and reverse weights, is assigned to each proposition.

In addition, each node carries an estimated cost corresponding to the difficulties that may confront a user wishing to provide the data needed for the proposition. Two control strategies that have been used by CASNET involve (1) the selection of the node with the maximum weight-to-cost ratio, and (2) selection of the node with the maximum weight subject to certain constraints on cost. Both these strategies tend to pick the node considered most likely true or most consistent with the current state of the network.

The control strategy presented by CASNET has several interesting advantages. First we notice that there is no depth-first traversal constraint. The system may pick and choose questions from any point in the network. As a result of this ability to move around, CASNET may search for the best question in the entire network. Thus, the framework provided by CASNET should allow a more intelligent control strategy than PROSPECTOR's.

It is not clear, however, that the specific heuristic applied in the calculation of node weights by CASNET is particularly optimal. An inference system designed to either prove or disprove a top proposition should weigh most strongly those propositions bearing the greatest influence on its top propositions. The consistency of a proposition with the rest of the givens should not be as critical to the control strategy as the ultimate influence of that proposition on the top proposition. Thus, it would be nice if we could develop a control strategy which searched the entire network for the proposition most likely to change the top proposition. In fact, this

technique has already been developed. The MULTIPLE program [7,8,9,10], utilizes a control strategy dependent on the cost-effective influence of each subnode on the top node or proposition.

3. The MULTIPLE Control Strategy.

MULTIPLE is an acronym for MULTIpurpose Program that LEarns. The original MULTIPLE program was designed to search a fairly general implicit proposition tree [7,8,9,10]. Implicit AND/OR trees for games and theorem proving are handled well by MULTIPLE. The program has the additional ability to learn through experience. MULTIPLE has been implemented in the domains of the game of Kalah and the resolution principle with promising results [11].

The MULTIPLE control strategy is really a best-first algorithm that efficiently selects the seemingly best proposition at any stage, to work on next. This is accomplished with a two step algorithm: first the system "sprouts" from the most meritorious untried proposition on the proposition tree. After sprouting, the merits generated for the newly sprouted propositions are backed up to the top proposition. At each level, only the best merit along with the proposition it represents is backed up, and finally at the top level the most meritorious untried proposition is found. By alternately: (1) sprouting from the most meritorious proposition, and (2) backing up merits, MULTIPLE always works on the proposition it considers most promising.

Assume, for example, that proposition G12 is the most meritorious untried subproposition in figure 1. The MULTIPLE program will sprout its descendants G121, G122, ...,G12n and pick the most

meritorious of these. The merit of that best subproposition, G12j is first backed up to G12. Next, the merit at G12 is compared to those merits previously stored at G11 and G13, the maximum merit being backed up to G1. Finally, the merits at G1 and G2 are compared and the best one is backed up to G. At this point we have identified a new most meritorious proposition and may start again.

Central to this entire procedure is the concept of merit. We now proceed to define this concept. Assume for a moment that we have a general proposition tree with a top proposition G and subpropositions Gi (for i = 1 to n). Each subproposition Gi may itself have subpropositions designated Gij (for j = 1 to m). In general, an additional subscript will indicate another level down the proposition tree. The merit of an untried proposition Gij...st is defined by the partial derivative:

$$\left| \frac{d\,P}{d\,C_{ij\ldots st}} \right| \qquad \text{DEFINITION OF MERIT} \qquad (3.1)$$

where dP is the change in the probability of the top proposition G, and dCij...st is the cost of expanding the untried proposition Gij...st. Absolute value is used because we do not differentiate between changes in probability in the positive or negative directions. What matters to the merit is the absolute ability of node Gij...st to influence the probability of proposition G if Gij...st is expanded.

Note that this definition of merit describes in precise mathematical terms those qualities we desire most for the next proposition on the inference network which is to be expanded. A high merit states that a proposition will exert much influence on the top proposition with little cost. Low merits indicate that expansion of a proposition will have little effect on probabilities at the top level or that the expansion will be accomplished only at a high cost.

The merit has been expressed as a derivative relating P, the change in probability of the top node, to the cost of expanding an untried proposition somewhere else on the proposition tree. Instead of expressing the derivative as such, we find it simplier to apply the chain rule and evaluate the derivatives of linked nodes .

$$\left| \frac{dP}{dCij\ldots st} \right| = \left| \frac{dp}{dPi} * \frac{dpi}{dPij} * \ldots * \frac{dPij\ldots s}{dPij\ldots st} * \frac{dPij\ldots st}{dCij\ldots st} \right|$$

DEFINITION OF MERIT (3.1)

The last factor in this expansion is the only one involving the cost of expanding the untried node. It is the self-merit of that proposition, and represents the ability to change the probability of the untried subproposition, per unit cost applied in expansion of that subproposition. For our purposes, we will approximate the self-merit by an expert opinion, and so we need not worry about calculating it.

$$\left|\frac{dP_{ij\ldots st}}{dC_{ij\ldots st}}\right| == \left|\frac{\Delta P_{ij\ldots st}}{\Delta C_{ij\ldots st}}\right| \qquad \text{APPROXIMATION OF SELF-MERIT}$$

The remaining factors of the merit involve the influence of the change in the probability of a subproposition on the probability of its immediate father. When dealing with inference networks, we shall refer to each of these factors in the merit formula as a link-merit. Every antecedent-consequent pair has its own link-merit. Thus, the link-merit may be thought of as being associated with the link from antecedent to consequent. A link-merit corresponds to the degree of influence exerted by an antecedent on its consequent. In practice, link-merits are calculated by differentiation of the functions used in the updating scheme from antecedent to consequent.

The most meritorious proposition on a proposition tree, is defined by MULTIPLE as the untried subproposition having the highest merit. This proposition is known to have the greatest potential for influencing the top proposition. In an inference network, such an unexpanded proposition has the greatest potential for influencing the top proposition.

The process of finding merits, it should be noted, is performed in a time proportional to the tree-depth. Only the merits on the newly expanded proposition need be computed for backing up. The other merits are already in place at each node that has been

previously traversed. This process is thus completely analogous to moving up a tree of winners. Execution time is proportional to tree-depth rather than tree-size. Thus, the merit values which are calculated to order a best-first traversal of the inference network, are themselves computed with a best-first strategy by the MULTIPLE algorithm. We now describe the MULTIPLE method for merit computation as it applies to inference networks. An optimal, albeit slightly slower, technique that uses an exhaustive depth-first traversal for finding merits is suggested in the conclusion.

MULTIPLE always applies its efforts on the most promising subproposition. This has proved to be a very effective technique in several domains. Apparently, the power of the techniques stems from the fact that it disregards those alternatives which do not appear promising. This resembles to a large extent the way an expert may approach a consulting problem. We have therefore decided to apply merits to the domain of expert consultant systems.

4. Merit In An Inference Network.

The concept of merit as presented in MULTIPLE is easily adapted for expert systems and inference network traversal. An expert system control strategy that consistently requests information only on the most pertinent proposition in the inference network, will ask the fewest questions in the long run. As we have seen, asking for the proposition of maximum merit is equivalent to asking the most pertinent question with respect to the top proposition. The most meritorious proposition in the network will be the proposition which is most influential on changes in the probability of the top proposition with respect to the cost of its own expansion. Thus, we have designed an expert control strategy based on merits.

Applying the MULTIPLE algorithm to inference networks, an expert system explores the propositions possessing the highest merits until it encounters a proposition marked as askable. The system then halts its traversal of the network to prompt the user for the appropriate information. After receiving that information or finding that the user is unable to supply it, the system proceeds to discover the next unasked, askable proposition of highest merit. The entire process is iterated until there are no more propositions to be found with a greater merit than some cutoff value.

When there are several top propositions the most meritorious node may be defined as that proposition with the highest merit in any

of the various networks stemming from these top consequents. This interpretation is equivalent to defining a new supernode that may be influenced by all the top propositions for purposes of merit propagation. Thus, handling an inference system with several top propositions is a simple extension of the single top consequent case, with the minor restriction that all top propositions be measured in similar units.

The cutoff merit is a parameter controlled by the user. It may be utilized to limit or increase the total number of questions asked, but will not alter the order of questioning. Therefore, there is no reason to restrict this value once traversal has begun. Rather, the user may change its value at any time to prematurely terminate the traversal, or extend it to the entire network. Only those propositions with merit above the cutoff will be asked. If no such propositions remain, then there is no purpose to be served by further traversal of the network, and we are done.

The MULTIPLE best-first algorithm we have presented will be superior to the depth-first procedure previously applied in most expert system control strategies. The merit system is not constrained to traverse the network in a set order as are depth-first strategies. Furthermore, the merits compared come from the entire network rather than just a set of nodes with a common father. Thus, the MULTIPLE mechanism for selecting the most meritorious proposition in the network should result in fewer questions than the corresponding depth-first strategy.

An objection, however, may be raised to the degree of jumping around on the inference network resulting from this best-first traversal. A depth-first algorithm, it may be argued, will remain within a single subtree for a length of time and never return to that region of the network again. The user will therefore be questioned thoroughly on one topic before questioning switches to another subtree. The merit based algorithm may jump all around the inference network in a sequence that is bewildering, and may result in confusing the user.

In reply to this objection, we note that a merit control strategy may actually be implemented within the constraints of a depth-first traversal of the inference network. Merits may be utilized to order the sons or antecedents of a node before it is expanded by the depth-first traversal. Merit values may be used as a uniform mechanism for prioritizing and perhaps even cutting off the antecedents of a node within the depth-first framework. We believe, however, that the time saved with the best-first plan of action far outweighs any potential disadvantage that may result from changing the order of questioning.

Furthermore, the freedom to alter the cutoff merit value for traversal of the inference network was a trivial matter with the best-first algorithm. Since the propositions are traversed in order of decreasing merit, the value of the cutoff merit does not influence which nodes are traversed, but only when the traversal should halt. Increasing or decreasing the cutoff only extends or limits the total

number of questions asked. Similar reasoning does not apply with a depth-first approach. Assume, for example, that the user began his session with a high cutoff value, and many antecedents or sons of nodes on the leftmost subtrees already traversed were pruned off. If he now chooses to decrease the cutoff value, the depth-first strategy offers no mechanism to return and evaluate those nodes. Once a node has been examined by a depth-first traversal, it is gone forever and never reexamined. Likewise, if the user starts with a low cutoff value, and later increases it, he will already have wasted time traversing many propositions in the early subtrees with low merit. Any changes in the cutoff merit for a depth-first traversal may apply only to parts of the tree not yet traversed. The MULTIPLE control strategy may be much more flexible here because all nodes are reconsidered for questioning before each question is asked.

5. Merits, Link-Merits, and Self-Merits.

How do we determine these magical quantities known as merits? This question should actually be divided into its two components. First we must be able to find self-merits, and then we need to calculate the link-merits. Link-merits and the self-merit along a path from any node to the top consequent are simply multiplied to provide the merit value of the node, as specified in equation 3.1.

Self-merit was defined as the change in probability for a proposition per unit cost of expanding or working on that proposition. To an expert familiar with the inference network setup, we assign the task of choosing self-merits. These need not be in any specific range, but should be correct relative to each other. A proposition whose parameters are easily specified by a user, and whose probability is likely to change a great deal will have a high value for dP/dC. Such a proposition should be granted a high self-merit value. Conversely, a proposition for which the user is unlikely to or slow at providing an answer, or which is rarely changed much in probability, should be assigned a low self-merit. Self-merits of unaskable nodes are proportional to the expected change in the node probability per unit cost of expanding the node to its immediate descendants.

Furthermore, we may define self-merits for various propositions that are not requested from the user, but either input

from a mechanical or electronic source or calculated by the computer. A calculation requiring only core space and little execution time has very high self-merit. Those propositions requesting access to a random storage device such as a disk, may have slightly lower self-merits. Finally, those propositions whose parameters may only be obtained from slower devices such as tape drives have even lower self-merits. Of course all of these self-merits are relative to the self merits assigned to nodes requiring user interface. Since users are generally slower than machines, a user related proposition may have even less self-merit.

Additional considerations may also apply within the realm of user related propositions. Some questions are harder to answer. These should have lower self merit, from the point of view of cost. An entire table of numbers, for example, is more difficult to input than a single yes/no answer. These cost factors must be weighed together with the chances that the response will change the proposition's probability to effectively determine self-merits.

Complications in self-merits also arise from the variation in the pool of users. One user may find it simplier to respond to questions of a specific type while others may have differing preferences. Thus, we may need several sets of self-merits for accurate merit calculation. All of these considerations must be weighed in the design of self-merits. The most important consideration of all, however, is that these self-merits be internally consistent throughout the inference network.

Following determination of the self merit, the remaining terms in the merit formula are all link-merits of the form dPi/dPij. These depend on the mathematical relationship between antecedents and consequents. The next several sections deal with the derivation of these link-merits. For most updating schemes, finding link-merits involves only a trivial amount of differentiation. With differentiation and variable substitution routines, this could even be done automatically.

6. "AND", "OR", "NOT" Link-Merits.

The probability of an antecedent Ej, as estimated by a user, we will term $P(Ej|Ej')$, following the notation of Duda et al [1], where Ej' are the relevant observations upon which it is based. A consequent whose truth is contingent upon the verification of all of its antecedents is the logical "AND" of those antecedents. In a more general probabilistic approach, assuming that all antecedents are independent, the AND link may be mathematically described by the equation:

$$P(H|E1',\ldots, En') = P(E1|E1') * \ldots * P(En|En') \quad \text{"AND" LINK} \quad (6.1)$$

where the ANDed probability of the consequent on the left, given the present probability of each antecedent Ej, is just the product of all current antecedent probabilities. The link-merit of the consequent with respect to any antecedent Ej may be found by calculating the partial derivative of the consequent probability, $P(H|E1',\ldots,En')$, with respect to the probability of that antecedent, $P(Ej|Ej')$. We now proceed to transform these link merits, first described in [8,9,10] to the Duda notation.

$$\frac{d\ P(H|E1',\ldots,En')}{d\ \ P(Ej|Ej')} = P(E1|E1') * \ldots * P(Ej-1|Ej-1') * P(Ej+1|Ej+1') * \ldots * P(En|En') \quad \text{AND-LINK-MERIT} \quad (6.2)$$

Noticing the similarity between the link-merit and the definition of ANDing, we may rewrite the AND-link-merit as:

$$\frac{d\ P(H|E1',\ldots,En')}{d\ \ P(Ej'|Ej')} = \frac{P(H|E1',\ldots,En')}{P(Ej|Ej')}$$

AND-LINK-MERIT (6.3)

This simplified form of the link-merit depends only upon the probabilities of the consequent and the antecedent under consideration. Such a form is very useful for actual computations, and we will therefore attempt to simplify all our link-merits to this format.

Sometimes a consequent is known to hold if any one of its antecedents is true. Such a node is said to be linked to its antecedents with the "OR" function. In mathematical terms, assuming independent antecedents, the OR link may be expressed by the function:

$$P(H|E1',\ldots,En') = 1 - [1 - P(E1|E1')] * \ldots * [1 - P(En|En')]$$

"OR" LINK (6.4)

where the consequent probability on the left hand side is the complement of the products of the complements of all antecedent probabilities. Applying the definition of link-merit to equation 6.4, we find that OR-link-merit may be specified by:

$$\frac{d\ P(H|E1',\ldots,En')}{d\ \ P(Ej|Ej')} = \begin{array}{l}[1 - P(E1|E1')] * \ldots * [1 - P(Ej-1|Ej-1')] * \\ [1 - P(Ej+1|Ej+1')] * \ldots * [1 - P(En|En')]\end{array}$$

OR-LINK-MERIT (6.5)

Employing the same type of substitution for the OR-link-merit as we applied to the AND-link-merit, the form may be simplified and expressed in terms of only the specific antecedent - consequent pair being considered. Substituting equation 6.4 into equation 6.5 we find:

$$\frac{d\ P(H|E1',\ldots,En')}{d\ P(Ej|Ej')} = \frac{[1 - P(H|E1',\ldots,En')]}{[1 - P(Ej|Ej')]} \quad \text{OR-LINK-MERIT (6.6)}$$

Equations 6.3 and 6.6 for the evaluation of link-merits are the actual forms used by MULTIPLE for the calculation of merits. The AND-link-merit as well as the OR-link-merit approach a finite limit as P(Ej|Ej') approaches 0 and 1 respectively. This may be observed in equations 6.2 and 6.5 where there is no chance of obtaining a zero in the denominator. Thus, these merit values are always defined.

Often it is convenient to classify a consequent as the negation of its antecedent. In logical terms, the consequent is true when its antecedent is false, and false when its antecedent is true. In a probabilistic scheme, such a consequent may be given the complement of its antecedent probability.

$$P(H|E') = 1 - P(E|E') \qquad \text{"NOT" LINK} \quad (6.7)$$

Note the absence of a subscript on the antecedent E. A NOT link has only one antecedent. Thus, merits are not needed to choose among the

sons of such a consequent. However, the link-merit of NOT links will be used in the MULTIPLE type of control strategy. It can easily be shown that the NOT-link-merit is -1:

$$\frac{d\ P(H|E')}{d\ P(E|E')} = -1 \qquad \text{NOT-LINK-MERIT} \qquad (6.8)$$

Apparently, the negative sign may be disregarded here since only absolute values are significant for merits. Thus, one might be led to conclude that a NOT link leaves unchanged the merits of its subpropositions. In section 11, however, we note that the sign on a merit value may be very significant during merit propagation in networks with multiple fathers on a single proposition.

Some consultant systems utilize the notion of fuzzy AND and OR nodes in the inference network. The probability of a fuzzy AND node is simply the minimum of all its antecedents probabilities, while that of a fuzzy OR node is their maximum. Obviously, these fuzzy links are not differentiable functions, and have no defined link-merits with our present scheme. We prefer the logical AND and OR type links because they use all antecedents in the process of updating. It is, however, possible to adapt the calculation of merits to fuzzy probabilities. This may enhance a system such as PROSPECTOR, which arbitrarily chooses its next question for propositions where the antecedent is constructed with the fuzzy AND and OR.

We propose two methods for the handling of fuzzy links. One possibility is to update the probabilities with fuzzy statistics but perform the merit calculations as if regular logical links had been used (equations 6.3 & 6.6). This should provide a fairly good approximation since both the AND antecedent and the OR antecedent exert the most influence on their consequent in both fuzzy and logical updating methods under similar conditions (see section 9). The second possibility would also involve updating with the fuzzy techniques, but would calculate link-merits for differentiable approximations to the fuzzy functions. The first of these two possibilities may be regarded as a special case of the second. The logical AND and OR are just used to approximate the fuzzy AND and OR in this first possibility.

7. "MYCIN" Link-Merits.

The MYCIN system, designed to assist physicians with the diagnosis of microbial infections, utilizes a model for inexact reasoning in medicine [5,6]. Proceeding under the assumption that medical reasoning is intuitive and not expressible in precise probabilistic terms, Shortliffe developed a rule-based inferencing scheme that updates with an informal reasoning process. This technique, although not formally based in statistics, presents an interpretation of probability based upon confirmation.

Two basic concepts, the measure of belief (MB) and measure of disbelief (MD) are defined for the relationship between all linked propositions on the inference network. The MB[H,E] is a measure of the belief in the consequent H, based on all available current evidence E. Similarly, the MD[H,E] is a measure of the disbelief in H given the present situation E. Mathematically, the measure of belief in a consequent H with respect to a specific antecedent E1 is expressed as the ratio of the increase in the belief of H motivated by the knowledge that E1 is true, to the maximum possible increase in the certainty of H. The measure of disbelief is similarly defined with respect to the increase in the disbelief in H. For any single antecedent to H, E1 for example, either MB or MD must be zero. An antecedent that, when proven true, increases the belief in H will have a positive measure of belief but zero measure of disbelief. Likewise, an antecedent whose truth diminishes the probability of its

consequent will have a positive measure of disbelief, but a zero measure or belief. A proposition E1, that influences H in no way whatsoever, has the property that $MB[H,E1] = MD[H,E1] = 0$.

A formal definition of these measures is give by:

measure of	case	
	$P[H\|H'] = 1$	1
Belief $MB[H,E1]$ ==	$P[H\|E1] <= P[H\|H']$	0
(eq. 7.1)	otherwise	$\dfrac{P[H\|E1] - P[H\|H']}{1 - P[H\|H']}$
	$P[H\|H'] = 0$	1
Disbelief $MD[H,E1]$ ==	$P[H\|E1] >= P[H\|H']$	0
(eq. 7.2)	otherwise	$\dfrac{P[H\|H'] - P[H\|E1]}{P[H\|H']}$

Extending the Duda notation which we have adapted, $P[H|H']$ is the current probability of the consequent H. $P[H|E1]$ is the probability of H given that antecedent E1 is known to be true.

Although the measures of belief and disbelief are updated individually, they are later combined to provide a certainty factor that is the difference of the two.

$$CF[H,E1] = MB[H,E1] - MD[H,E1] \qquad (7.3)$$

A certainty factor is calculated for the antecedent of each link in MYCIN's inference network. Antecedent certainty factors may be used to update the measures of belief and disbelief in the consequent. The process of discovering an antecedent certainty factor may be arbitrarily complicated since an antecedent may itself consist of any number of propositions in conjunction or disjunction and these propositions may themselves depend upon other antecedents. Measures of belief and disbelief are calculated for each proposition in the antecedent of a consequent to be updated. These measures are combined with each other under the rules of fuzzy logic, to find the total measures of belief and disbelief on the antecedent. The certainty factor of the antecedent is determined by combining these measures, and is then used to update the consequent.

This final inferencing step, the inexact method for updating of consequents, is of particular interest here. Each antecedent may be linked to a consequent with a rule describing the maximum measures of belief or disbelief in the consequent, denoted MB'[H,E1] and MD'[H,E1], given that the antecedent is absolutely believed. The new antecedent E1 may then update its consequent H in the current situation E, by increasing the consequent measure of belief:

$$MB[H,E\&E1] = MB[H,E] + MB'[H,E1] * CF[E1,E] * (1 - MB[H,E]) \qquad (7.4)$$

if $MB'[H,E1] > 0$, or by increasing the measure of disbelief:

$$MD[H,E\&E1] = MD[H,E] + MD'[H,E1] * CF[E1,E] * (1 - MD[H,E]) \qquad (7.5)$$

if $MD'[H,E1] > 0$. Since either MB[H,E1] or MD[H,E1] will be zero in every case, only one of the two updating equations will be applied in any one case. Equation 7.4 is used in the case of confirmation of supporting evidence, while equation 7.5 updates the hypothesis according to evidence that tends to decrease its plausibility.

The MYCIN system has several complicating features and special cases which are applied to these rules. We shall derive the link merit for a simplified version of the MYCIN scheme. If all rules are assumed to increase the belief in their consequents, and endpoint conditions are ignored, then updating may be expressed as:

$$CF[H,E\&E1] = CF[H,E] + CF'[H,E1] * CF[E1,E] * (1 - CF[H,E]) \qquad (7.6)$$

where CF'[H,E1] is the maximum certainty in H gained from the knowledge that E1 is absolutely true. In our simplification, the increase in consequent certainty with respect to the change in antecedent certainty may be expressed as:

$$\frac{d\ CF[H,E\&E1]}{d\ CF[E1,E]} = CF'[H,E1] * (1 - CF[H,E]) \qquad (7.8)$$

This quantity is the link-merit on our simplified MYCIN updating rule. An actual link merit for the real MYCIN link may be calculated with similar reasoning and the use of a differentiable approximation to the MYCIN updating scheme.

The link-merit we have found for MYCIN links indicates that in a depth-first traversal of the inference network, disregarding the costs of expanding antecedents, the antecedent with the greatest CF'[H,E] should be the first one expanded. This mathematical analysis supports Shortliffe's suggestion for dynamic ordering of rules by certainty factors and expansion costs [5,6]. Introducing self-merits into our analysis would, of course, provide a much more rigorous test for prioritizing the antecedents in a MYCIN style inference network. The self-merit of an antecedent may just be approximated by a value inversely proportional to the number of propositions on which it directly depends. Such a MULTIPLE type, merit based scheme would surely improve the efficiency of MYCIN.

8. Subjective Bayesian "EVIDENCE" Link-Merits

Duda, Hart, and Nilsson [1] have introduced a subjective Bayesian updating method which relates a consequent to its antecedents by a function which we associate with the EVIDENCE link. The EVIDENCE link model is developed with the assumption of conditional independence of the antecedents both under the consequent and under the negation of the consequent. Pednault, Zucker, and Muresan have shown that with the additional assumption of mutually exclusive and exhaustive consequents, the EVIDENCE updating scheme breaks down [4]. In general, however, the EVIDENCE updating scheme is able to function. We first present a synopsis of that method.

From Bayes rule we know that:

$$P(H|Ej') = \frac{P(Ej'|H) * P(H)}{P(Ej)} \qquad P(-H|Ej') = \frac{P(Ej'|-H) * P(H)}{P(Ej)} \tag{8.1}$$

The probability of a consequent, given its antecedent Ej with some current probability, is equal to the probability that the antecedent will be at its current probability given the consequent, multiplied by the prior probability of the consequent, and divided by the prior probability of the antecedent.

We define the relationship between probabilities and odds as:

$$O = \frac{P}{1 - P} \qquad (8.2) \qquad\qquad P = \frac{O}{1 + O} \qquad (8.3)$$

An effective likelihood ratio @j' (read lambda sub j), is defined as:

$$@j' = \frac{P(Ej'|H)}{P(Ej'|-H)} = \frac{O(H|Ej')}{O(H)} \quad \text{EFFECTIVE LIKELIHOOD RATIO} \quad (8.4)$$

The two forms of the effective likelihood ratio may be shown to be equivalent with equation 8.1 and either equation 8.2 or equation 8.3.

Duda et al [1] describe a method for calculating P(H|Ej') through linear interpolation. For each antecedent of H, a graph of P(H|Ej') vs. P(Ej|Ej') is plotted (figure 2). To plot this graph one must obtain two points: a probability for H given the antecedent, P(H|Ej), and a probability for H under the negation of the antecedent, P(H|-Ej). In addition, the Duda method utilizes prior probabilities for the consequent, P(H), as well as the antecedent, P(Ej).

Finding the approximation to P(H|Ej') with the graph in figure 2 is just a simple linear interpolation. Given a value for P(Ej|Ej'), the antecedent probability, we may interpolate with the following two equations to find P(H|Ej'), the predicted consequent probability.

case	formula
$P(Ej \mid Ej') < P(Ej)$ then	$P(H \mid Ej') = P(H \mid -Ej) + P(Ej \mid Ej') * \dfrac{P(H) - P(H \mid -Ej)}{P(Ej)}$
(8.5)	
$P(Ej \mid Ej') >= P(Ej)$ then	$P(H \mid Ej') = P(H) + [P(Ej \mid Ej') - P(Ej)] * \dfrac{P(H \mid Ej) - P(H)}{1 - P(Ej)}$
(8.6)	

For n antecedents to a consequent H, the odds on H may be updated with the expression :

$$O(H \mid E1', \ldots, En') = \left[\prod_{i=1}^{n} @i' \right] * O(H) \qquad (8.7)$$

given the assumption that each antecedent Ej is independent of all the rest.

The careful reader will notice that we have now developed a sequence of mathematical steps that will allow the updating of $P(H \mid E1', \ldots, En')$, given some new values for the Ej's. These steps are arranged in table 1.

Now, proceeding to the task at hand, we must find the EVIDENCE-link-merit for the Subjective Bayesian updating method. As with all other link-merits, calculating the EVIDENCE-link-merit is just a matter of computing the derivative:

$$\frac{d\ P(H|E1', \ldots, En')}{d\ P(Ej|Ej')} \qquad \text{EVIDENCE-LINK-MERIT} \quad (8.11)$$

Employing the chain rule of differentiation, and noting the various dependencies of the terms in table 1 upon one another, we may express the EVIDENCE-link-merit derivative as:

$$\frac{d\ P(H|E1',\ldots,En')}{d\ O(H|E1',\ldots,En')} * \frac{d\ O(H|E1',\ldots,En')}{d\ O(H|Ej')} * \frac{d\ O(H|Ej')}{d\ P(H|Ej')} * \frac{d\ P(H|Ej')}{d\ P(Ej|Ej')}$$

CHAIN RULE FORM OF EVIDENCE-LINK-MERIT (8.11)

The following argument will produce the mathematical simplification of the above form of EVIDENCE-link-merits. Those readers not interested in the derivation, should skip to the end of this section for the final result. Before attempting to evaluate equation 8.11 we shall digress for the moment and compute some useful partial derivatives. These derivatives will be used in the evaluation of the chain rule form of equation 8.11 to produce a simplified form of the EVIDENCE-LINK-MERIT.

From equations 8.2 and 8.3 relating probabilities and odds we note that one may calculate the derivatives of each the probability and odds function with respect to the other.

$$\frac{dP}{dO} = \frac{1}{(1+O)^2} = \frac{P^2}{O^2} = (1-P)^2 \qquad (8.12)$$

$$\frac{dO}{dP} = (1+O)^2 = \frac{O^2}{P^2} = \frac{1}{(1-P)^2} \qquad (8.13)$$

Next, let us attempt to evaluate the partial derivative of the logarithm of the updated odds with respect to the odds predicted by antecedent Ej. Recall equation 8.9 from table 1:

$$\frac{d \ln O(H|E1', \ldots, En')}{d \quad O(H|Ej')} = \frac{d\, [(1-n) * \ln O(H) + \sum_{i=1}^{n} \ln O(H|Ei')]}{d\, O(H|Ej')}$$

where $\sum$ is the summation of its argument over all i for i = 1 to n. Noting here that all terms in the numerator, with the exception of O(H|Ej') are constants with respect to O(H|Ej'), we can express this derivative as:

$$\frac{d \ln O(H|E1', \ldots, En')}{d\ O(H|Ej')} = \frac{d \ln O(H|Ej')}{d\ O(H|Ej')} = \frac{1}{O(H|Ej')}$$

Applying the chain rule, and rearranging terms, we have:

$$\frac{d \ln O(H|E1', \ldots, En')}{d\ O(H|E1', \ldots, En')} * \frac{d\ O(H|E1', \ldots, En')}{d\ O(H|Ej')} = \frac{1}{O(H|Ej')}$$

$$\frac{d\ O(H|E1', \ldots, En')}{d\ O(H|Ej')} = \frac{O(H|E1', \ldots En')}{O(H|Ej')} \qquad (8.14)$$

One last derivative which we must analyze before returning to the equation for EVIDENCE-link-merits is: d P(H|Ej') / d P(Ej|Ej'), the last term in equation 8.11. Note that P(H|Ej') is a function of P(Ej|Ej') through linear interpolation as given in equations 8.5 and 8.6. The derivative for each 8.5 and 8.6 with respect to P(Ej|Ej') are different but they are both just equivalent to the slope of the interpolation graph. Let us call this slope Mj for simplicity.

Note that the two slopes corresponding to Mjl and Mjr in figure 2 are not equal. Furthermore, at the point P(Ej|Ej') = P(Ej) there are two valid slopes. Which value should we use for Mj ? This question will be answered in section 10. For the duration of this discussion, we will assume that Mj is a known value.

We are now in a position to evaluate the EVIDENCE-link-merit as expressed in equation 8.11. Substituting the various partial derivatives which we have calculated for the terms in equation 8.11 as it was expressed after applying the chain rule, we find that the EVIDENCE-link-merit may be written as:

$$\frac{1}{[1 + O(H|E1',\ldots,En')]^2} * \frac{O(H|E1',\ldots,En')}{O(H|Ej')} * [1 + O(H|Ej')]^2 * Mj$$

$$\frac{d\ P(H|E1',\ldots,En')}{d\quad P(Ej|Ej')} = \left[\frac{1 + O(H|Ej')}{1 + O(H|E1',\ldots En')}\right]^2 * \frac{O(H|E1',\ldots,En')}{O(H|Ej')} * Mj \tag{8.15}$$

Equation 8.15 will allow us to compute the link-merit of any evidence link. Furthermore, it can be shown that this merit value is always defined. Recall from equation 8.9 that $O(H|E1', \ldots,En')$ may be expressed as the product of all the various effective likelihood ratios for the various antecedents, and the prior odds on the consequent. The effective likelihood ratios are themselves just ratios of the predicted consequent odds, $O(H|Ei')$, to the prior consequent odds, $O(H)$, for $i = 1$ to n. With the exception of $O(H|Ej')$, all the terms in equation 8.9 are constant with respect to the antecedent Ej. Using the constant C in place of these constant terms, we may write that:

$$O(H|E1', \ldots, En') = C * O(H|Ej') \qquad (8.16)$$

where

$$C = \left[\prod_{i=1}^{n} \frac{O(H|Ei')}{O(H)} \right] \quad \text{for all } i <> j$$

substituting this expression for $O(H|E1', \ldots, En')$ into 8.15,

$$\frac{d\ P(H|E1', \ldots, En')}{d\ \ P(Ej|Ej')} = \left[\frac{1 + O(H|Ej')}{1 + C * O(H|Ej')} \right] * C * Mj \qquad (8.17)$$

Thus, as $O(H|Ej')$ approaches zero, the limit of the EVIDENCE-link-merit approaches the finite value C * Mj. Similarly, as $O(H|Ej')$ approaches one, the limit of the EVIDENCE-link-merit approachen Mj / C.

Now that the EVIDENCE-link-merit has been shown to be defined in all cases, we will derive a more intuitive form of the expression, in terms of probabilities. Applying equation 8.3 to equation 8.15, we find that:

$$\frac{d\, P(H|E1', \ldots, En')}{d \quad P(Ej|Ej')}$$, the EVIDENCE-link merit is equal to

$$\left[\frac{[1 - P(H|E1',\ldots,En')]}{[1 - P(H|Ej')]} \right]^2 * \frac{P(H|E1', \ldots, En') * [1 - P(H|Ej')]}{P(H|Ej') * [1 - P(H|E1', \ldots, En')]} * Mj$$

cancelling appropriate terms leaves us with:

$$\frac{d\, P(H|E1', \ldots, En')}{d \quad P(Ej|Ej')} = \frac{[1 - P(H|E1',\ldots,En')]*P(H|E1',\ldots,En')}{[1 - P(H|Ej')] * P(H|Ej')} * Mj$$

EVIDENCE-LINK-MERIT (8.18)

This expression for the EVIDENCE-link-merit is the form actually employed in our implementation of the merit control strategy. In order to prevent any division by zero in the implementation of equation 8.18, the value of P(H|Ej') is offset by a small amount when it is found to equal zero or one. Once a value for Mj is determined, the remaining calculations are straightforward. The calculation of Mj, however, does pose some difficulty; this problem is addressed in section 10 when the merit control strategy is compared to the PROSPECTOR method.

9. An Intuitive Understanding of Merits

What do link-merits really mean, and why should we work on the proposition with the highest merit value? By definition, a high merit indicates a great chance of changing the top proposition probability. Each link-merit describes the power of an antecedent to change the probability of its direct consequent. Since changing consequent probabilities is the basic purpose of inference networks, it seems reasonable to choose the merit function as a priority rating. In previous sections we derived several important forms of the link-merit involved in merit calculation. The purpose of this section is to provide the reader with an intuitive understanding of why the link-merits appear in the forms we have derived.

The AND-link merit states that the power of an antecedent to change its hypothesis probability is inversely proportional to that antecedent's probability (equation 6.3). Thus, the antecedent of lowest probability has the highest link-merit among all the sons of an AND fact. Disregarding self-merits for the moment, we should always work on the least probable son of an AND node first. This is clear intuitively since the antecedent of lowest probability is the one primarily responsible for holding down the consequent probability of an AND link.

The OR-link-merit states that the power of an antecedent to change its consequent probability is inversely proportional to the

complement of that antecedent's probability. (equation 6.6). Thus, the antecedent of the highest probability has the greatest link-merit among all the sons of an OR node. Contrary to our findings with the AND proposition, if we disregard self merits for the moment, we should always work on the most probable son of the OR node first. This is also easily rationalized, since it is the antecedent of highest probability that primarily supports the consequent probability in an OR link.

Interestingly enough, the EVIDENCE-link-merit, as derived in equation 8.18, is similar to the product of the link-merits for the AND and the OR links described by equations 6.3 and 6.6, if we think of $P(H|Ej')$ as somehow related to $P(Ej|Ej')$. This relationship was initially quite surprising to us, and has provided us with several insights into the meaning of subjective Bayesian updating. We are tempted to view the EVIDENCE link as some combination of AND and OR links, or as a compromise between them.

Actually, we may note from equation 8.18 that antecedents predicting either a very high or very low consequent probability, $P(H|Ej')$ appear to exert the greatest influence on EVIDENCE links. Antecedents that predict consequent probability near .5 are not very influential on the actual consequent probability. Thus, an antecedent that tends to provide a very high or low consequent updating has greater EVIDENCE-link-merit and should be explored before its brothers.

Note that with AND links, the antecedent that wished to keep down the consequent probability the most, had highest merit. With OR links, the antecedent that was primarily responsible for keeping the consequent probability up earned the highest merit. An EVIDENCE link may earn merit through its attempts to either raise or lower the consequent probability away from .5. It thus seems to be a combination of the AND and OR type links.

This analysis offers us an insight into the purpose of EVIDENCE links. AND links should be used when the full power of the AND is needed to reduce consequent probability to very low values. OR links serve the purpose of allowing consequent probabilities to increase to near unity. EVIDENCE links are best used when the consequent probability should vary symmetrically around its prior probability. EVIDENCE links have a symmetrical updating ability, combining aspects of AND link updating with properties of OR links.

Our analysis and comparison of AND, OR and EVIDENCE links has centered primarily on the denominator in the link-merit terms. These denominators discriminate among the merits of the various antecedents for a specific consequent. The numerators in all these link-merit formulas, however, are also quite important when the propositions being compared are from arbitrary places on the inferences network and not just the antecedents of one consequent. In that case, merit values must be calculated with the complete formulas as they have been derived. These values may then be compared for any two propositions on the network no matter how they are related.

Thus, to utilize the merit functions for ordering sons in a depth-first traversal, only parts of the link-merit functions need be considered. This provides a trivial calculation for the prioritizing of sons in such a traversal. The MULTIPLE algorithm, however, will require use of the actual link-merit that is derived through differentiation since merits from various dissimilar links are compared.

10. Validity of EVIDENCE-link-merits

The PROSPECTOR system developed at SRI International implements a depth-first traversal of the inference network. At each EVIDENCE node the antecedents are ordered with the MARK IV strategy. This strategy, described in the PROSPECTOR report [2], depends upon the J* function values assigned to each antecedent. Duda et. al. have designed the J* function to favor antecedents that tend to increase consequent probability when the consequent probability is low, and to favor antecedents that decrease consequent probability when the consequent probability is high.

We decided to test our merit function against J* on several real cases of probability updating. The J* function was programmed as specified in the PROSPECTOR report. The EVIDENCE-link-merit function used for comparison purposes was the one from equation 8.18. However, before presenting the results, we must explain the Mj values in that equation. From equations 8.5 and 8.6 it is apparent that Mj has two different values depending on whether $P(Ej|Ej')$ is greater or less than $P(Ej)$. These correspond to the two slopes in Figure 2.

The derivative techniques employed for deriving merits is correct for infinitesimal changes. Thus, if $P(Ej|Ej') < P(Ej)$ and we use the slope from the left side of the plot in Figure 2, Mj1, to compute the merit, that merit will be correct for probability changes that take place completely on the left half of the plot.

Equivalently, merits computed where $P(Ej|Ej') > P(Ej)$ and using the Mjr value, will be correct for probability changes that take place on the right half of the plot.

What shall we do for the point $P(E|Ej') = P(Ej)$? Apparently, that point has a left link-merit and a right link-merit, corresponding to the left and right link-merit derivatives. Furthermore, any $P(Ej|Ej')$ point close to $P(Ej)$ may also have a probability change that forces the antecedent probability to pass over $P(Ej)$. Would it be proper to compute link-merits for those points as if only infinitesimal changes in the antecedent probability will take place ?

Our solution to this problem employs an effective slope for Mj that is some combination of the two slopes Mjl and Mjr. In selecting a function to combine left and right slopes, we applied two constraints. They are: (1) it is reasonable to expect the effective slope at $P(Ej|Ej') = P(Ej)$ to be the average of $|Mjl|$ and $|Mjr|$, and (2) as a one moves further to the left, Mjl should quickly become the dominating slope; likewise a move to the right should result in a heavier weight to Mjr. We decided that a logarithmic growth and exponential decay for the weights applied to the left and right slopes would produce an acceptable continuous approximation to Mj. This approximation is used for all values of $P(Ej|Ej')$, including the point $P(Ej|Ej') = P(Ej)$. Since our probability changes will be finite rather than infinitesimal, it would not be proper to use either Mjl or Mjr even at the endpoints.

Given all the above constraints, the following approximation was developed:

The slopes for the left and right interpolation lines on figure 2 may be shown to have the form:

$$Mjl = \frac{P(H) - P(H|-Ej)}{P(Ej)} \quad (10.1) \qquad Mjr = \frac{P(H|Ej) - P(H)}{1 - P(Ej)} \quad (10.2)$$

An approximation to the slope may be produced by combining the two slopes Mjl and Mjr with weight factors C1 and C2.

$$Mj = C1 * |Mjl| + C2 * |Mjr| \quad (10.3)$$

Finally, the weighting function to determine the constants C1 and C2:

Case 1 -

if P(Ej|Ej') <= P(Ej) then

$$C1 = .5 + \frac{\ln\left[1 + K * \frac{P(Ej) - P(Ej|Ej')}{P(Ej)}\right]}{2 * \ln(1 + K)}$$

$$\text{and} \quad C2 = 1 - C1 \quad (10.4)$$

Case 2 -

if P(Ej|Ej') >= P(Ej) then

$$C2 = .5 + \frac{\ln\left[1 + K * \dfrac{P(Ej|Ej') - P(Ej)}{1 - P(Ej)}\right]}{2 * \ln(1 + K)}$$

$$\text{and} \quad C1 = 1 - C2 \qquad (10.5)$$

The values of C1 and C2 are always in the interval [0,1]. Furthermore, when P(Ej|Ej') = P(Ej), both equations 10.4 and 10.5 reduce to the value of .5, giving both Mjl and Mjr equal weight. Thus, the proposed equations seem to satisfy our constraints. The only unknown remaining is the constant K introduced in the weighting functions, which determines how quickly the weighting function changes as the point P(Ej|Ej') moves. A larger value for K will cause the weight of the slope on the side to which P(Ej|Ej') moves to increase more quickly. Using the arbitrarily selected value of K = 10 we tested the merit function vs. the PROSPECTOR J* function.

A more mathematically rigorous scheme might utilize a parabolic approximation for the interpolation process, providing a differentiable function and a simple technique for computing Mj. A parabolic approximation would also obviate the requirement for using absolute values on the slopes of Mjl and Mjr. As we shall see in the

next section, the sign of the merit function can be significant. It would therefore be advantageous to keep the sign of the interpolation slope. In fact, even with our linear interpolation technique it might be best to keep the signs of Mjl and Mjr as long as both terms are of the same sign.

Consider the simple proposition tree in Figure 3. The top consequent on the tree is H, and the two antecedents are called E1 and E2. The method of subjective Bayesian updating is employed to propagate changes in the antecedent probabilities, P(E1|E1') and P(E2|E2'), to the consequent P(H|E1', E2'). Two links are defined, one from each antecedent to H. For each link we set P(H|E) = 0.9 and P(H|-E) = 0.1, so that both sons have similar updating power. The prior probability at each node, P(H), P(E1), and P(E2) is set to 0.5, self-merits are set to 1, and the antecedent probabilities are varied individually. Results of this test are shown in Table 2.

Our intuitive analysis of the EVIDENCE link-merit functions described in Section 8 is substantiated by the test data. The link-merit tends to be maximized for an antecedent if it updates the consequent probability away from 0.5 toward 0 or 1. The link-merit from E1 to H, for example, increases as P(E1|E1') moves away from .5 and E1 attempts to update the consequent toward a more extreme probability. This line of reasoning obviously applies only to the various antecedents of a single consequent, when their influences on consequent probability are compared to each other. However, it does substantiate the general claims that the subjective Bayesian method

provides a more symmetrical updating mechanism than ANDing or ORing, and that it should be applied when the user is equally interested in the variation of consequent probability in both directions from is prior status.

Furthermore, it should be apparent by noting the changes in P(H|E1',E2'), the updated consequent probability, that the link-merit function for EVIDENCE type propositions is a maximum for the antecedent that actually bears the most influence on the consequent probability. If we wish to select the potentially most influential antecedent for a specific consequent, the merit value provides a superior heuristic to the J* function of PROSPECTOR.

11. Merit Propagation in Inference Networks.

When applied to inference networks, merit propagation is often beset with special classes of problems not handled by the MULTIPLE algorithm previously defined. Inference networks, due to their generalized graph structures, present special situations not present in an ordinary proposition tree. In an inference tree where no node has more than one father or consequent, determination of merits may proceed precisely as defined for the MULTIPLE system. In many inference networks, however, a node may have several fathers corresponding to an antecedent with several consequents. Such situations present special problems in the backing up of merits.

Suppose, for example, that we have a proposition tree in which the two sons, G1 and G2, of a top proposition G *have a common* subproposition G' among their various antecedents (Figure 4). Assume further that G' is found to be the most meritorious descendant of both G1 and G2 independently. That is to say that the merit value at G' is greater than the merit backed up at either G11 or G22. In this case, G' will be chosen as the most meritorious node over all of its brothers, and its merit will be passed up to both G1 and G2. The merit calculated coming down the left pathway from G to G1, to G' will be backed up to G1 along the left pathway , while the merit of the right pathway from G to G2, to G' is backed up to G2 along the right pathway.

A dilemma arises when the merits at G1 and G2 are compared for backing up. It would not be accurate to back up the maximum of these two merits as is usually done by MULTIPLE, since either choice represents the selection of the same subproposition G'. We must instead back up to G a merit corresponding to the combined effects of G' through the left and right paths. Adding the absolute magnitudes of the backed-up merits at G1 and G2 would also not be proper, however, since the effects of G' through its left and right fathers may tend to cancel each other rather than be additive. It is possible that G' exerts a positive influence through G2. Thus, the most appropriate course of action when backing up merits from G1 and G2 would be to add their signed merit values. If both branches influence the top proposition G in a similar direction, their effect on the magnitude of the merit will be additive. If, however, G1 tends to increase the probability of G and G2 tends to decrease it, their merits will be of opposite signs, and the total merit will be diminished.

This solution to the problem of multiple consequents has a firm mathematical basis. Recall from the initial definition of the merit function as a product of derivatives, that the link-merits from G to G' are of the form:

$$\frac{d\,P(G)}{d\,P(G1)} * \frac{d\,P(G1)}{d\,P(G')} \quad \text{left-links} \quad (11.1) \qquad \frac{d\,P(G)}{d\,P(G2)} * \frac{d\,P(G2)}{d\,P(G')} \quad \text{right-links} \quad (11.2)$$

G2 for purposes of further propagation, the merit originally from G11 may be greater than that merit at G2, and may therefore be discovered as the most meritorious node. This may, however, be a drastic error; the combined merit from both paths from G' to G may have been greater than the merit backed up from G11.

Let us analyze the origin of this problem. While backing up from G11 and G' to G1, we considered only the merits present at those propositions, and decided that the merit of G11 was greater. If we would have considered the consequences of combining the effects of the various paths out of G', we might have reached a different conclusion. However, since we know only about the merit values in the subpropositions of G1 when updating G1, there does not seem to be any way we might have avoided this dilemma. It thus appears that there is a potential propagation error with any proposition having multiple fathers that is not chosen as the most meritorious son by all of its fathers.

Several solutions to this problem are possible, but none of them is perfect. One may fist be tempted to assign a node extra merit for having additional fathers. This will tend to select a multifather proposition over its unifather brothers. This extra weighting, however, is not always desirable. A proposition may exert opposing influences through its various parents. Such a node will have a lower, rather than greater effect on the probability of a top proposition. Thus, it is certainly not clear that a multifather proposition deserves greater merit than its unifather brothers.

We are interested in the value of the total merit derivative d P(G) / d P(G'), however, rather than the individual left and right influences. Noting that P(G) is just a function of P(G1) and P(G2), that P(G1) is a function of P(G11) and P(G'), and that P(G2) is a function of P(G') and P(G22), we may apply the chain rule for functions of several variables:

$$\frac{d\,P(G)}{d\,P(G')} = \frac{d\,P(G)}{d\,P(G1)} * \frac{d\,P(G1)}{d\,P(G')} + \frac{d\,P(G)}{d\,P(G2)} * \frac{d\,P(G2)}{d\,P(G')} \qquad (11.3)$$

This formula many be extended to allow the calculation of merit for any number of propositions with a common descendant.

Furthermore, although we have not bothered to mention the propagation of self-merits in this discussion, they present no additional difficulty. In general, self-merits of the leaf proposition are multiplied into the link merits when the process of backing up begins. A trivial application of the distributive law to equation 11.3 allows implementation of that algorithm in this case.

$$\frac{d\,P(G)}{d\,P(G')} * \frac{d\,P(G')}{d\,C(G')} = \frac{d\,P(G)}{d\,P(G1)} * \frac{d\,P(G1)}{d\,P(G')} * \frac{d\,P(G')}{d\,C(G')} + \frac{d\,P(G)}{d\,P(G2)} * \frac{d\,P(G2)}{d\,P(G')} * \frac{d\,P(G')}{d\,C(G')} \qquad (11.4)$$

Thus, when the merit of G' or any other proposition with more than one father is backed up, the self-merit may be combined with the link merits as usual. Any time a group of brothers is compared for backing up to their father, we must check and see whether their merits are of a common origin. For unrelated merits, we simply back up the merit of maximum absolute magnitude as we would in a normal tree structure. Merits of common origin, however, must be added before the backup.

Allow us to consider a slightly more complicated example. Suppose now that our previous proposition tree has n subpropositions at the top level (Figure 5). Propositions G1 and G2 share a common descendant G, but the remaining brothers G3...Gn have no common descendant or are unexpanded. Merits are backed up for all the subpropositions G1 to Gn, and must now be backed up to G. However, before they can be backed up, the merits from G1 and G2 must be combined since they have a common origin for their merit values. Thus, our general procedure for backing up will be to first check all brothers for common merit origins. Those with merits of common origin are combined with merit addition, and only then is the maximum merit value propagated up the proposition tree.

A more severe problem results in the proposition tree of Figure 4 when G' is not chosen as the most meritorious descendant of both G1 and G2. Suppose, for example, that G1 has a more meritorious son G11, and the merit of G11 is backed up to G1 instead of the merit of G'. When that merit is compared to the merit backed up from G' to

A more stable solution might involve the backing up of all the merits from multifather propositions at each node on the way up. This way, if there is any merit that is combined with a merit of common origin during the back up it will be identified. Perhaps a more pragmatic approach would be to back up the K-best merits and hope for the best.

The most practical approach to merit propagation in inference networks may just be to ignore the problem caused by multifather nodes. After all, the objective is to save time by choosing the proper proposition to ask the user about at each point. If more time is wasted finding that best proposition than by using a slightly less meritorious one, the entire purpose of the intelligent control strategy has become self-defeating. Thus, unless a more efficient mechanism for handling multiple fathers in an inference network is discovered, we believe that the present MULTIPLE algorithm provides the best control strategy to date.

12. DISCUSSION: The Generality of Merit Functions

We have shown that the application of merits, first developed for the MULTIPLE system, to inference networks allows efficient updating of consequent propositions. Link-merit, which we have derived for several types of antecedent-consequent associations, is simply a mathematical function representing the ability of an inference rule to alter their consequent's probability. Self-merit, approximated by the expert, is the ratio of the expected change in our belief in a proposition to the cost of expanding that proposition. The total merit of a node on the network is the product of the link-merits on all the links directed from that node to a top proposition, multiplied by the self-merit of that node. Because the merit of any proposition on the network is a measure of the cost effective ability of that proposition to change a top proposition, a control strategy that selects the most meritorious proposition for questioning is acting in an intelligent manner.

The units in which merit expresses the cost effective ability of any sub-proposition to influence one of the top propositions are universal to all propositions in an inference network linked to a common consequent. This equivalence of the units used to express the merit is a property of the merit function, and is true regardless of the types of links used, or the units in which the lower level propositional plausibilities are expressed. The merits of a MYCIN style proposition and a PROSPECTOR evidence node located in the same network will be expressed in equivalent units. All merit values in an

inference network are expressed in units equal to those used in a top consequent, divided by cost. The merit based control strategy is therefore applicable to networks with any mixture of propositional types. One restriction introduced by merits, however, is that the belief in all top consequents be measured in similar units.

This versatility of the merit function allows the merit control strategy to operate with various types of propositions in the same network. Suppose, for example, we have a proposition in our system called "NUMERICAL- SUPERIORITY". "NUMERICAL-SUPERIORITY" is a function of the number of elements in items A and B, and thus has two antecedents referred to as "ITEM=A=SIZE" and "ITEM=B=SIZE" respectively, which may be actual numbers rather than probabilities. "NUMERICAL-SUPERIORITY" is related to its antecedents by the function:

$$\text{NUMERICAL-SUPERIORITY} = \frac{\text{ITEM=A=SIZE} - \text{ITEM=B=SIZE}}{\text{ITEM=A=SIZE} + \text{ITEM=B=SIZE}} \tag{12.1}$$

the link-merit from ITEM=A=SIZE to NUMERICAL-SUPERIORITY is derived as:

$$\frac{d\ \text{NUMERICAL-SUPERIORITY}}{d\ \text{ITEM=A=SIZE}} = \frac{2 * \text{ITEM=B=SIZE}}{(\text{ITEM=A=SIZE} + \text{ITEM=B=SIZE})^2}$$

substituting for ITEM=B=SIZE with equation 12.1, we find:

$$\frac{d\ \text{NUMERICAL-SUPERIORITY}}{d\ \ \text{ITEM=A=SIZE}} = \frac{(1 - \text{NUMERICAL-SUPERIORITY}^2)}{2 * \text{ITEM=A=SIZE}}$$

a similar calculation may be performed to determine the link-merit of ITEM=B=SIZE.

A generalized inference network may be updated with propositions whose plausibilities are stored in many forms. The linking functions described by the rules that construct the network will need to take this into account when updating consequents. The merit formulas, however, will always be found with the same algorithm. Therefore, it is reasonable to assume that a computer may be programmed to derive link-merits on an inference network for which it is supplied with the linking formulas.

We are now ready to present our vision of a future expert consultant system. Propositions, supplied by an expert, will be linked into an inference network by linking functions, also specified by the expert. Common updating functions such as NOTing, ANDing, and ORing of antecedents, as well as the MYCIN scheme for inexact reasoning, and the method of subjective Bayesian updating would be system defined links. The expert may employ these predefined functions in his links, or proceed to define his own set of linking

functions. Once the system has created the network specified by the expert, it will derive the various functions required for link-merit calculation. Differentiation and variable substitution routines will be available to the system for merit calculation on any new expert defined linkage functions. If a new function is found to be useful it may be stored in the data base of common link types for future use. This future system will allow the expert complete flexibility in creating the network, and free him from the burdensome calculations that may be needed for finding the link-merits.

Furthermore, for the sake of completeness we should point out that a merit based best-first traversal may be applied to inference networks in which the propositions or rules contain variables. An example of a variable rule is "if there is evidence for the presence of organism x, then initiate treatment for organism x". Variables may be very helpful in limiting the number of propositions and rules that must be instantiated in the computer's memory for specific cases. An algorithm for handling such variables has been developed for the MULTIPLE program's implementation of the resolution principle in theorem proving [11].

13. CONCLUSIONS

Expert consultant systems have shown their adaptability to many important problems. These systems have incorporated a valuable tool, the inference network, in the analysis of various top consequents. An inference network consists of propositions ordered into a graph structure to allow propagation of information from lower level, simpler, propositions to the more esoteric top proposition of the network. The majority of time consumed by the inferencing process is needed for questioning of the user. A significant reduction in the numbers of propositional parameters filled in by the user will markedly reduce execution time and increase the cost effectiveness of expert consultants.

We have applied the concept of merit, first developed for use in the MULTIPLE system, to inference networks. Merit, a function of both the cost and potential benefits of expanding a proposition, is a quantity easily calculated by computers. The MULTIPLE algorithm prioritizes the propositions under consideration by their merits. In an inference network, the merit may direct an intelligent traversal of the propositions, and an efficient ordering of questions to be asked from the user.

The askable proposition of maximum merit on an inference network corresponds to the parameter having the largest potential cost effective influence on some top consequent, and should be

expanded before working on propositions of lower merit. Introduction of a cutoff merit may allow termination of user questioning when there remain no unknown parameters which may significantly alter the top consequent probability. Such a questioning strategy will minimize the time needed by the user to answer insignificant questions, and increase the efficiency of the inferencing process.

In this paper we have explicitly shown the derivation of link-merits for "AND", "OR", "NOT", "MYCIN", and "EVIDENCE" type links in an inference network. The techniques utilized in these derivations, however, may be applied to any other type of link representing a differentiable function. New classes of links developed by experts designing inference networks should be adaptable to the best-first approach based on merit calculations. Perhaps in some future system computers may even be programmed to derive the link-merits for an inference network. A system with such a capability could easily utilize the merit mechanism for network traversal by combining the link-merits which it would derive with the expert supplied self-merits at the startpoints of the propagation. The expert designing an inference network would never be required to deal with merit functions or their derivation, but just specify the links as mathematical functions, and provide a self-merit for each proposition on the network.

The concept of merit provides a flexible and useful tool in the design of control strategies for expert consultant systems. Merit values may be employed to order antecedents prior to the expansion of

a depth-first traversal, or they may themselves direct a best-first MULTIPLE type of inference network traversal. We believe that the MULTIPLE algorithm will offer a significant saving of time over the classical depth-first approach. The original MULTIPLE algorithm was designed for implementation with indefinitely large trees such as those created when proving theorems or playing games. An exhaustive search of such a tree is not realistically feasible. With finite inference networks, however, it may be possible to save additional time by a more exhaustive system for the updating of merits. When searching for the unasked, askable proposition of maximum merit in the inference network, an expert system may first perform an exhaustive depth-first merit analysis, extending and expanding the network traversal at each proposition until it reached an askable proposition. The endpoints or leaves on this exhaustive merit analysis would include all the askable propositions under consideration by the system at that time. Such a mechanism for merit propagation would examine all the possible askable nodes, and discover the absolutely optimal proposition for questioning the user.

Thus, we propose two possible implementations for the introduction of a merit based best-first control strategy. The first scheme involves calculation of merit values with the MULTIPLE algorithm, always expanding the most meritorious descendant. The merit values which guide the user through the inference network in a best-first traversal, would also be calculated in a similar manner. This technique offers the advantage of quick merit calculation since the time for the traversal needed to calculate the merit values is

proportional to the depth of the inference network. A second possible implementation for inference network control strategies is to calculate the merit values with an exhaustive depth-first network traversal. This might require slightly more time for finding the most meritorious proposition on the network, but it would guarantee that the user is always questioned on the absolutely most meritorious proposition.

REFERENCES

1. Duda, R.O., Hart, P.E., and Nilsson, N.J. Subjective Bayesian Methods for Rule-based Inference Systems. PROC. National Computer Conference 45, AFIPS Press, pgs. 1075-1082. Artificial Intelligence Center, SRI Int., Menlo Park, CA., 1976.

2. Duda, R.O., Hart, P.E., Konolige, K., and Reboh, R., A Computer-based Consultant for Mineral Exploration. Artificial Intelligence Center, SRI Int., Menlo Park, CA., (Sept. 1979).

3. Melle, W.V., A Domain-independent System that Aids in Constructing Knowledge-based Consultation Programs. Dept of Computer Science, Stanford Univer., Stanford, CA., (June, 1980).

4. Pednault, E.P.D., Zucker, S.W., and Muresan, L.V., On the Independence Assumption Underlying Subjective Bayesian Updating. Artificial Intelligence 16 (May, 1981), pgs. 213-222.

5. Shortliffe, E.H., and Buchanon, B.G., A Model of Inexact Reasoning in Medicine. Mathematical Biosciences 23 (1975), pgs. 351-379.

6. Shortliffe, E.H. Computer-based Medical Consultations: MYCIN, American Elsevier, New York, 1976.

7. Slagle, J.R. Artificial Intelligence: The Heuristic Programming Approach. McGraw-Hill, New York, 1971.

8. Slagle, J.R. A Multipurpose Theorem-proving Heuristic Program that Learns. Proc. IFIP Conf., 1965, vol. II., Spartan books, Wash, D.C., pgs. 323-328.

9. Slagle, J.R. and Bursky, P. Experiments with a Multipurpose Theorem-proving Heuristic Program. Journal of the Assoc. for Computing Machinery, 15, (January, 1968), pgs. 85-89.

10. Slagle, J.R. and Farell, C.D. Experiments in Automatic Learning for a Multipurpose Heuristic Program. Comm.. of the ACM, 14, (February, 1971), pgs. 91-99.

11. Slagle, J.R. and Koniver, D.A. Finding Resolution Proofs Using Duplicate Propositions In AND/OR Trees. Information Sciences, 3, (January, 1971), pgs. 315-342.

12. Weiss, S.A., Kulikowski, C.A., Amarel, S. and Safer, A. A Model-based Method for Computer-aided Medical Decision Making. Artificial Intelligence 11, (August, 1978), pgs. 145-172.

APPENDIX

The j-star function we have used for comparison with merits in section 9 is described in [2]. For the sake of completeness, we summarize that description:

Let

$$LS = \frac{O(H|Ej)}{O(H)} \qquad LN = \frac{O(H|-Ej)}{O(H)} \qquad L' = \frac{O(H|Ej')}{O(H)}$$

where O(H) are the prior odds on H, O(H|Ej) are the odds on H given that Ej is true, O(H|-Ej) are the odds on H given that Ej is false, and O(H|Ej') are the odds on H given the present odds on Ej.

Define the measures of belief and disbelief in a similar manner to that used for the MYCIN system in section 4.

$$MB(H|E') = \begin{cases} \dfrac{P(H|Ej') - P(H)}{1 - P(H)} & \text{if } P(H|Ej') > P(H) \\ 0 & \text{otherwise} \end{cases}$$

$$MD(H|E') = \begin{cases} \dfrac{P(H) - P(H|Ej')}{P(H)} & \text{if } P(H|Ej') < P(H) \\ 0 & \text{otherwise} \end{cases}$$

We may now define j* as a function of these terms:

Case 1 -

if LS > LN

$$J^* = \frac{LS}{L'} * P(Ej|Ej') * [1 - MB(H|Ej')] + \ln \frac{L'}{LN} * [1 - P(Ej|Ej')] * [1 - MD(H|Ej')]$$

Case 2 -

if LS < LN

$$J^* = \frac{LN}{L'} * P(Ej|Ej') * [1 - MD(H|Ej')] + \ln \frac{L'}{LS} * [1 - P(Ej|Ej')] * [1 - MB(H|Ej')]$$

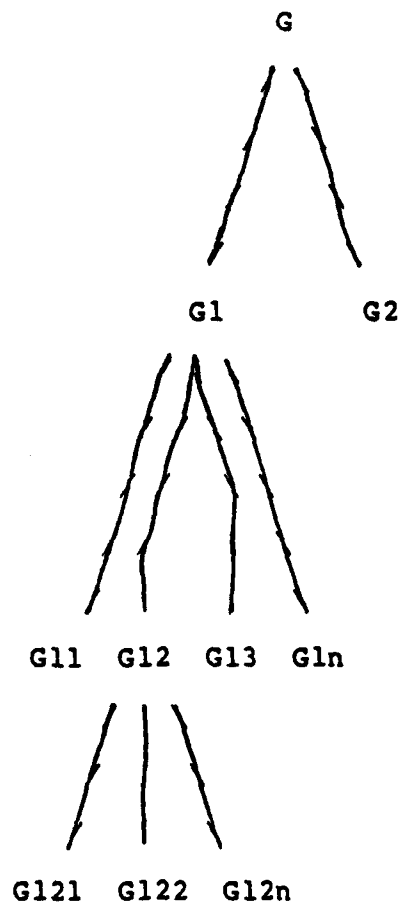


Figure 1. A proposition tree. G is the top level proposition. Each node Gij...st is assigned a merit based upon its ability to influence G and the cost of sprouting from it.

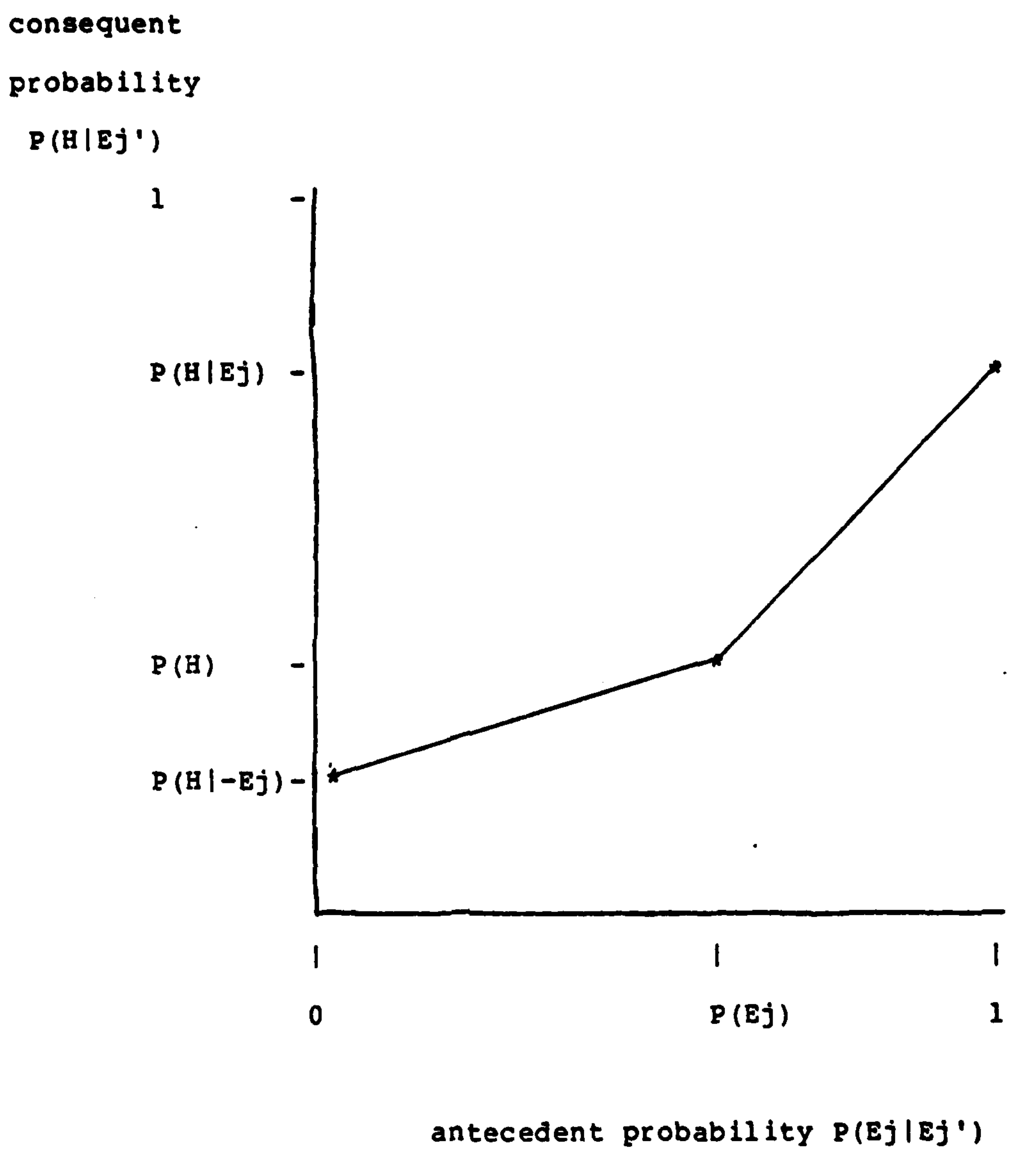


Figure 2. A plot of consequent probability, P(H|Ej'), vs. antecedent probability, P(Ej|Ej'). One such plot is interpolated for each antecedent of a consequent to be updated. Straight lines are used between the three points for interpolation. The slope of the line on the left, between 0 and P(Ej) is called Mjl, while the slope of the second half of the line between P(Ej) and 1 is called Mjr.

Table 1. Steps in Subjective Bayesian updating.

Step 1 -

Linear interpolation is used to find P(H|Ej') for each Ej

If P(Ej|Ej') <= P(Ej) then (equation 8.5)

$$P(H|Ej') = P(H|-Ej) + P(Ej|Ej') * \frac{P(H) - P(H|-Ej)}{P(Ej)}$$

If P(Ej|Ej') >= P(Ej) then (equation 8.6)

$$P(H|Ej') = P(H) + [P(Ej|Ej') - P(Ej)] * \frac{P(H|Ej) - P(H)}{1 - P(Ej)}$$

Step 2 -

Predicted consequent probabilities are converted to odds using equation 8.2. P(H|Ej') is the predicted probability for the consequent H, considering only the current probability for the antecedent Ej'.

$$O(H|Ej') = \frac{P(H|Ej')}{1 - P(H|Ej')} \qquad (8.8)$$

table 1 (continued)

Step 3 -

Effective likelihood ratios of the antecedents are combined to determine the current odds on the consequent H. Note that this step is contingent upon the independence of the various antecedents. (results of combining equations 8.4 and 8.7)

$$O(H|E1', \ldots, En') = \left[\prod_{i=1}^{n} @i' \right] * O(H) = \left[\prod_{i=1}^{n} \frac{O(H|Ei')}{O(H)} \right] * O(H) \qquad (8.9)$$

Step 4 -

Odds of consequent are converted back to probabilities using equation 8.3. The value P(H|E1', ...,En') is the final updated probability for the consequent H.

$$P(H|E1', \ldots, En') = \frac{O(H|E1', \ldots, En')}{1 + O(H|E1', \ldots, En')} \qquad (8.10)$$

Figure 3. The simple tree used for testing the EVIDENCE-link-merit function against J*. P(H) = P(E1) = P(E2) = .5 and the probabilities at E1 and E2, P(E1|E1') and P(E2|E2') are independently varied. Results are presented in table 2.

Table 2. Comparison of EVIDENCE-link-merit and J* Functions

			Link-merit from H to:		J*-Function from H to:	
P(E1|E1')	P(E2|E2')	P(H|E1',E2')	E1	E2	E1	E2
.50	.50	.500	.800	.800	2.197	2.197
.40	.50	.420	.800	.780	1.953	2.197
.30	.50	.340	.800	.718	1.588	2.197
.20	.50	.260	.800	.616	1.128	2.197
.10	.50	.180	.800	.472	0.592	2.197
.01	.50	.108	.800	.308	0.062	2.197
.01	.40	.081	.615	.243	0.062	1.953
.015	.40	.084	.617	.252	0.092	1.953
.005	.40	.078	.614	.235	0.031	1.953
.01	.405	.082	.624	.246	0.062	1.968
.01	.395	.079	.607	.241	0.062	1.937
.30	.70	.500	.891	.891	1.588	1.588
.30	.90	.701	.747	1.136	1.588	0.592
.40	.90	.767	.586	.968	1.953	0.592
.41	.90	.773	.573	.951	1.983	0.592
.42	.90	.779	.560	.934	2.012	0.592
.45	.90	.795	.525	.883	2.035	0.592

table 2. (continued)

.39	.90	.761	.600	.985	1.921	0.592
.38	.90	.755	.614	1.002	1.889	0.592
.35	.90	.736	.659	1.052	1.784	0.592
.40	.91	.777	.569	.973	1.953	0.535
.40	.92	.787	.551	.979	1.953	0.478
.40	.95	.816	.492	.996	1.953	0.302
.40	.89	.758	.603	.962	1.953	0.648
.40	.88	.748	.619	.957	1.953	0.704
.40	.85	.720	.663	.941	1.953	0.868

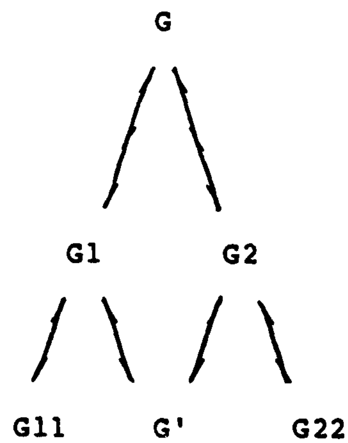


Figure 4. A proposition tree. G is the top node. G' is a subproposition of both G1 and G2. Thus, G' influences G through both a left and a right path. How should we propagate the merit of G' ?

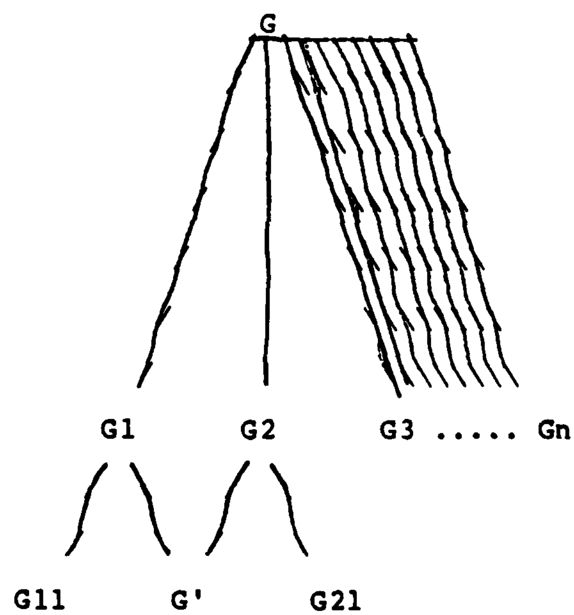


Figure 5. A proposition tree in which two of the subpropositions share a common descendant but the other subpropositions have independent children. We must treat the merit propagated from G' in a special way, but use the normal propagation routine for the other propositions.

DAT
FILM